UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/760,405	01/12/2001	Xiaoning Nie	GR 00 P 1031	1766

24131 7590 06/30/2004

LERNER AND GREENBERG, PA
P O BOX 2480
HOLLYWOOD, FL 33022-2480

EXAMINER	
O BRIEN, BARRY J	

ART UNIT	PAPER NUMBER
2183	

DATE MAILED: 06/30/2004

9

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 09/760,405	**Applicant(s)** NIE, XIAONING	
	Examiner Barry J. O'Brien	**Art Unit** 2183	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *13 April 2004*.

2a)☒ This action is **FINAL**. 2b)□ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-12* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-12* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) □ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application (PTO-152)

6) □ Other: _____.

DETAILED ACTION

1. Claims 1-12 have been examined.

Papers Submitted

2. It is hereby acknowledged that the following papers have been received and placed on record in the file: Amendment A as received on 4/13/04.

Specification

3. The lengthy specification has not been checked to the extent necessary to determine the presence of all possible minor errors. Applicant's cooperation is requested in correcting any errors of which applicant may become aware in the specification.

4. The applicant is requested to review the specification and update the status of all co-pending applications made mention of, replacing attorney docket numbers with current U.S. application or patent numbers when appropriate.

Claim Rejections - 35 USC § 112

5. The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

6. Claim 11 is rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the enablement requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to enable one skilled in the art to which it pertains, or with which it is most nearly connected, to make and/or use the invention.

Claim 11 recites the limitation, "a process is called by assigning a process number, a priority, and a memory address of a starting point of the process in the program memory". However, the specification only describes a process being called by executing a RUN instruction, with the process number, priority and memory address being operands of the RUN instruction (see p.15 of the specification), which is different than a process being called as a result of assigning a process number, a priority and a memory address.

Claim Rejections - 35 USC § 102

7. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (a) the invention was known or used by others in this country, or patented or described in a printed publication in this or a foreign country, before the invention thereof by the applicant for a patent.

8. Claims 1-8 and 10 are rejected under 35 U.S.C. 102(a) as being anticipated by Gupta et al., U.S. Patent No. 5,941,983.

9. Regarding claim 1, Gupta has taught a data processing device for processing in parallel a plurality of independent processes, comprising:

a. A program memory (78 of Fig.3) having stored therein at least one compiled program with a multiplicity N of independent processes, the compiled program including information on parallelism and a multiplicity of bundles with a plurality of instructions of a process (see Col.6 lines 4-14),

b. A branching control unit (80 of Fig.3) connected to and addressing said program memory (see Col.9 lines 44-52). Here, because the instruction fetch units are fetching instructions from the instruction queues, they are inherently addressing the program memory in order to read out instructions from it.

c. A register (88 of Fig.3) for storing flags and data which are switched in dependence on a process being executed (see Col.8 lines 57-60 and Col.9 lines 62-65),

d. A program flow control unit (80 of Fig.3) connected to said branching control unit, said program flow control unit controlling a fetching of bundles from said program memory and said branching control unit and an output of instructions in dependence on information contained in the instructions and included in a compiling time of the program (see Col.9 lines 44-52).

10. Regarding claim 2, Gupta has taught the data processing device according to claim 1, which comprises a number N instruction buffers (see 94a-c of Fig.3) connected in parallel downstream of said program memory for storing instructions read out from said program memory (see Fig.3).

11. Regarding claim 3, Gupta has taught the data processing device according to claim 2, which comprises an instruction output selector (80 of Fig.3) connected to and controlled by said program flow control unit such that said instruction output selector reads out instructions from said instruction buffers and outputs N instructions in parallel (see Col.9 lines 44-62).

12. Regarding claim 4, Gupta has taught the data processing device according to claim 1, which comprises N instruction decoders (80 of Fig.3) for decoding the instructions being output (see Col.9 lines 44-52).

13. Regarding claim 5, Gupta has taught the data processing device according to claim 1, comprising at least two instruction-execution units (84 of Fig.3) for outputting the N decoded instructions.

14. Regarding claim 6, Gupta has taught the data processing device according to claim 5, which comprises a data memory (86 of Fig.3) and at least two buses connecting said N instruction-execution units to said data memory (see connections between 84 and 86 of Fig.3).

15. Regarding claim 7, Gupta has taught the data processing device according to claim 1, wherein said program flow control unit is configured to execute the instructions of one or more bundles in parallel (see Col.1 line 55 – Col.2 line 9, Col.6 lines 39-55, and Col.10 lines 20-26).

16. Regarding claim 8, Gupta has taught the data processing device according to claim 1, wherein said branching control unit is configured to output an address pointer for addressing a bundle (see Col.9 lines 44-52). Here, because the instruction fetch units are fetching instructions from the instruction queues to be executed in parallel using instruction pointers (see Col.20 lines 22-33), and because they are fetching multiple independent instructions simultaneously using the instruction pointers, they are inherently outputting instruction pointers to address the multiple independent instructions (bundles).

17. Regarding claim 9, Gupta has taught the data processing device according to claim 1, but has not explicitly taught wherein the branching control unit comprises:

a. A first multiplexer and a second multiplexer. While not taught explicitly, the function of two multiplexers is inherent in the operation of the circuit (see paragraphs (e), (f), and (g) below).

b. An adder. Here, Gupta has taught the use of a PC counter distance encoded in an instruction to update the PCs for each individual thread (see Col.20 lines 22-33). Because a PC counter distance or an absolute new address can be encoded for branch instructions (see Col.20 lines 39-51), as well as the next sequential fetch address as long as the encoded dependencies for non-branch instructions are satisfied (see Col.6 lines 51-55 and Col.10 lines 6-26), there is inherently an adder so that the offset, whether branch offset or a sequential offset, can be added to the threads PC to create a new target address.

c. N program counters. Here, Gupta has taught each thread having its own instruction queue with corresponding program counter (see Col.6 lines 4-14 and Col.20 lines 26-32),

d. Wherein said program flow control unit feeds a number of instructions in a bundle to said adder and said adder adds an address pointer and the number of instructions. Here, Gupta has taught

e. Wherein said program flow control unit feeds addresses for program jumps or function calls and a process number to said first multiplexer. Here, Gupta has taught the location in each thread that a branch instruction may jump to being encoded in an instruction, and then updating the PC associated with each thread with this new PC value (see Col.20 lines 22-

32). Therefore, the circuit is inherently operating as a de-multiplexer, selecting a PC to write the jump address into based upon the thread the instruction is part of.

f. Said first multiplexer writing either the output signal of said adder or the addresses for program jumps or function calls into said program counter assigned to the active process. Here, Gupta has taught the instruction location in each thread that a branch instruction may jump to being encoded in an instruction, and then updating the PC associated with each thread with this new PC value if needed (see Col.20 lines 39-51), or an offset to a PC value may be used which requires the output of the adder as described above (see Col.20 lines 22-32). Therefore, the circuit is inherently operating as a de-multiplexer, selecting a PC to write the jump address, whether absolute or an offset, into based upon the thread the instruction is part of.

g. A content of said program counter assigned to the currently active process is output as a new address pointer via said second multiplexer which is controlled using the process number supplied. Here, Gupta has taught that multiple threads can be active at once, that each thread has its own PC and corresponding functional unit, thus allowing parallel execution of threads (see Col.6 lines 4-14 and Col.20 lines 22-32). Therefore, the circuit described by Gupta inherently operates as a multiplexer, selecting and outputting a PC to be executed from based on its associated thread.

18. Regarding claim 10, Gupta has taught the data processing device according to claim 1, wherein said program flow control unit is configured to receive via a subbus of an output bus of said program memory at least one of the following:

a. At least one bit for indicating the parallel execution of instructions,

b. At least one bit for indicating the length of the following instruction bundle,

c. The indication of one or more NOPs in the instruction bundles,

d. A priority of the processes of the instructions.

19. Here, Gupta has taught the encoding of dependency information within instructions, with the dependency information being a number of bits identifying a dependent instruction by giving its location (see Col.10 lines 6-19 and Fig.8). Because this information identifies instructions that cannot be executed in parallel, these bits can be considered to be an indication of an instructions ability to execute in parallel with another instruction, provided that it is not encoded as a dependent instruction. Because the claim is written in the alternative format, and Gupta meets one of the four options, the claim language is satisfied.

Claim Rejections - 35 USC § 103

20. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

21. Claim 11 is rejected under 35 U.S.C. 103(a) as being unpatentable over Gupta et al., U.S. Patent No. 5,941,983 as applied to claim 1 above, and further in view of Ito et al., U.S. Patent No. 5,742,782.

22. Regarding claim 11, Gupta has taught the data processing device according to claim 1, but has not explicitly taught wherein a process is called by assigning a process number, a priority and a memory address of a starting point of the process in the program memory.

23. However, Ito has taught the selection of a process using an instruction stream number (see "STN" of Fig.13 and Col.10 lines 50-51), a priority based on whether a NOP instruction is referenced in the instruction stream (see "Signal155" of Fig.13 and Col.11 lines 9-13, 21-24), and the address of the thread starting point (see Col.10 lines 31-41) in order to more efficiently schedule threads and more effectively use a processor's parallel processing capabilities (see Col.3 lines 32-36). One of ordinary skill in the art would have recognized the desire in microprocessor design to make a design more efficient, thus saving important processor resources. Therefore, one of ordinary skill in the art would have found it obvious to modify the processor of Gupta to include the selection of a thread based on assigning it a process number, a priority, and providing a memory address of the beginning of the process in order to more efficiently and effectively use the processor's parallel processing capabilities.

24. Claim 12 is rejected under 35 U.S.C. 103(a) as being unpatentable over Gupta et al., U.S. Patent No. 5,941,983 as applied to claim 1 above, and further in view of Allen, Jr. et al., U.S. Patent No. 6,404,752.

25. Regarding claim 12, Gupta has taught the data processing device according to claim 1, but has not explicitly taught wherein said data processing device is a network processor for processing layer 1 to 7 of protocol stacks in applications including LAN, ATM switches, IP routers, and frame relays based on a system selected from the group consisting of DSL, Ethernet, and cable modems.

26. However, Allen, Jr. has taught the use of general-purpose microprocessors as network processors to provide a cost-effective solution to processing protocol stack layers for IDSN, cable and DSL modems, that provides high throughput and speeds (see Col.1 lines 44-49 and Col.2 line 38 – Col.3 line 23). One of ordinary skill in the art would have recognized that a primary goal in microprocessor design is to lower costs while maintaining a high level of performance. Therefore, one of ordinary skill in the art would have found it obvious to modify the processor of Gupta to be used in a network processor because Gupta has taught a high-performance, general-purpose processor (see Col.5 lines 62-65), thus allowing costs to be kept low while providing a high level of network processing performance.

Response to Arguments

27. Applicant's arguments filed on 4/13/04 have been fully considered but they are not persuasive.

28. On page 15 of the present amendment, the Applicant argues in reference to claim 1, in essence:

> *"However, during execution of the program, the processor in GUPTA "issues instruction from each queue in sequential order" (Col.6 lines 15-16) and not "in parallel" as recited in claim 1 of the instant application."*

29. In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e., issuing instructions in parallel) are not recited in the rejected claim(s). Claim 1 recites a device "processing in parallel", but does not specify the issuing of instruction in parallel. Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

30. Furthermore, Gupta has in fact taught the issuing of instructions in parallel. Although Gupta has taught instructions being issued from each instruction queue sequentially as the Applicant has pointed out, instructions are issued, and subsequently processed, in parallel, with each of the three instruction queues issuing instructions in parallel to each of the three execution units (see Gupta, Col.6 lines 4-14 and Col.14 lines 20-67).

31. On pages 14 and 15 of the present amendment filed 4/13/04, the Applicant further argues with respect to claim 1, in essence:

> *"The GUPTA reference discloses a method for executing instruction out of order to improve performance of a processor. The method includes compiling the instructions of a single program into separate queues along with various encoded dependencies between instructions in the different queues...In contrast, the present invention as claimed in independent claim 1, provides a data-processing*

> *device wherein the data processing device or processor can carry out/process at least two independent processes (or threads) in parallel."*

32. However, Gupta has in fact taught a processor that can "carry out/process at least two independent processes (or threads) in parallel". Gupta has taught that the processor can use its queues for separate threads that operate in an environment that allows for multiple threads of execution (see Col.6 lines 4-7). Further, Gupta has taught that alternatively, the dependency information does not have to be encoded for dependencies between instructions in different queues, but can be for dependencies between instructions in the same queue, which would allow the threads in the separate queues to be independent (see Col.6 lines 12-14 and Col.18 lines 45-56).

33. On pages 17-19 of the present amendment filed 4/13/04, the Applicant argues with respect to claim 11, in essence:

> *"The ITO reference disclosed a processing apparatus for executing a plurality of VLIW threads in parallel. The processing apparatus simultaneously executes multiple threads of long instructions, where each thread is made up of multiple operations instructions. ... ITO does not assign "a process number, a priority, and a memory address ... in the program memory. Rather ITO develops these signals during execution and changes the cited signals during the course of execution. ... Even assuming, arguendo, that the "STN" is a process number and that "signal155" or TNC does represent a priority, ITO does not call a process "by assigning a process number, a priority, and a memory address ... in the program memory" as recited in claim 11 of the instant application".*

34. However, Ito has taught where a thread is selected (and subsequently "called"/executed) after the determination of a process number, a priority and a memory address. In Ito, the instruction stream number is a thread ID of the thread to be executed (see "STN" of Fig.13 and Col.10 lines 50-51), which is updated to a thread that is desired to be switched to (see Col.11 lines 32-38) upon determination of a priority based on whether a NOP instruction is referenced in the instruction stream (see "Signal155" of Fig.13 and Col.11 lines 9-13, 21-24), thus giving preference (priority) to threads which do not contain NOPs. Upon determination of which thread is to be executed, the corresponding first instruction of the thread is selected for execution (see Col.11 lines 32-38) and fetched from main memory (21 of Fig.1) (see 135 of Figs.1-3 and Col.6 lines 48-54).

35. On pages 19-21 of the present amendment file d4/13/04, the Applicant argues with respect to claim 12, in essence:

> "*However, the combinatin of ALLEN and GUPTA would require using mulitiple network processors, where each processor may process a single process in parallel. Unfortunately, this would require a tremendous amount of processing overhead to coordinate parallel processing and indicates that there is not a reasonable expectation of success for the proposed combination. In contrast, the configuration described in claim 12 of the instant application handles mulple processes in parellel in easch procesor and could easily be used in a network processor configuration, because there is no need to corordinate the idseprsal betwen the units.*"

36. However, Gupta has taught a data-processing device for processing in parallel a plurality of independent processes (see paragraph 15 of previous office action mailed 1/13/04, as well as above paragraphs 9 and 28-32). Allen has taught using multiple processors in a network processor for processing layer 1 to 7 of protocol stacks in parallel (see paragraphs 30-32 of previous office action mailed 1/13/04, as well as above paragraphs 24-26) that allow data to be processed in parallel, thus improving speed and throughput (see Allen, Col.2 line 62 – Col.3 line 12). Because the processor of Gupta has the advantage that it reduces cycle time and improves processor performance (see Gupta, Col.5 line 62 – Col.7 line 23), but has not taught its use as a network processor, one of ordinary skill in the art would have found it obvious to modify the processor of Gupta to be one of the multiple processors in a network processor as in Allen, so as to then provide a network processor with the ability to have each processor process a plurality of independent processes in parallel, as well as the ability to processor layers 1 to 7 of protocol stacks in parallel. Furthermore, because Allen has already taught the network processor being comprised of multiple general-purpose processors allowing for parallel processing (see Allen, Col.7 lines 16-39), the necessary hardware to coordinate parallel processing is already present, and thus one could reasonably expect success in the combination.

Conclusion

37. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

38. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. Applicant is reminded that in amending in response to a rejection of claims, the patentable novelty must be clearly shown in view of the state of the art disclosed by the references cited and the objections made. Applicant must also show how the amendments avoid such references and objections. See 37 CFR § 1.111(c).

39. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Barry J. O'Brien whose telephone number is (703) 305-5864. The examiner can normally be reached on Mon.-Fri. 6:30am-4:00pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (703) 305-9712. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

40. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Barry J. O'Brien
Examiner
Art Unit 2183

BJO
6/28/2004

Eddie C

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100